Project Proposal

Arana Charlebois, Tristan
Carvalho, Cam
Corbett, Cole
Kassie, Turner
Pauls, Andrew
Wallace, Jordan

Cosc 4P02
Dr. Naser Ezzati-Jivan

January 12, 2025

1 Group, Meetings, and Repository

Arana Charlebois, Tristan - Developer, back-end and front-end
Carvalho, Cam - Developer, back-end
Corbett, Cole - Developer, back-end
Kassie, Turner - Scrum Master and Developer, back-end
Pauls, Andrew - Product Owner and Developer, back-end
Wallace, Jordan - Developer, front-end
Meetings: Tuesdays and Fridays, 10:00am to 11:00am.
Github Repository
URL:
https://github.com/BrockU-4P02-Logistics-System/Logistics-Application

2 Introduction

Established logistics corporations have seen more and more growth due to the transition of shopping becoming an online experience, as well as the widespread availability of anyone to sell goods online. With an increase in home deliveries, there are more drivers for each company, leading to organizational issues for each company, specifically finding the optimal way to arrange these employees. Logistic companies should aim to optimize the number of drivers and routes that such drivers follow to improve finances and environmental impact. This has led us to aim to develop a software system capable of finding the optimal routes for travel, based on the nodes placed on a map and a given number of travelers, a so-called "route optimization service".

3 Description and Goals

Ideally, we want to produce a functional software system that has practical use for individuals and companies, specifically one that can determine the optimal route(s) based on the number of travelers and a given list of nodes on a map. On the back end, this would include deploying an algorithm like the traveling salesperson or the multi-traveling salesperson, determining the optimal route for each traveler, that considers factors such as visiting all nodes and minimizing overall distance traveled. On the front end, we would like to have a UI that is similar in nature to Google Maps, or OpenStreetMap, and allow for a user to specify stops either by listing them or by placing the nodes on the map. Another main feature of the system that we aim to develop is the capability for one to adjust the route based on a given metric, specifically the overall time to complete the cycle or the most fuel-efficient route, as both would be of significance to users. We aim to have a system that can function properly and efficiently, even when many nodes are specified by a user, and many travelers are being used, furthering the practicality for logistic services.

A software system that is effective in determining such routes is both of financial and environmental significance, both for companies and users alike. Finding the most fuel efficient routes promotes reducing emissions, for conventional cars, as well as saving a cars battery for electric cars, both ways of lowering one's gas or energy expenses. Finding effective delivery routes could also be seen as a way for companies to improve customer satisfaction, as more deliveries will be completed earlier if an optimized route is given.

One concern that we have for the project is the potential for large inputs, particularly, computing the optimal routes, when the number of nodes and travelers selected by users becomes very large. Traveling salesperson is very costly, so naturally, with more inputs, the system will become slower. Also, in practice, such a system should offer real-time updates, as traffic and route conditions will naturally change during the day, reducing the accuracy of the initial results.

4 Software Engineering Process

As a team, we will focus on following the agile development process, specifically, planning sprints and delivering the overall system in multiple portions. It allows for more efficient development as it seems reasonable to complete the back-end of the system, mainly the logic for finding routes, and then moving onto completing the interface and front end of the system. The safety and security of the system, while important, is not the main focus, instead, we are concerned about the overall performance and usability of the system. This calls for constant feedback, and having the option to adapt is a main benefit of agile development.

Another reason that an agile development process will prove to be beneficial is that the testing of the system can be done continuously, and this will help us ensure that each unit is properly developed, allowing for more efficient integration with other components, rather than completing the system, and then testing, as we want to be sure that the logic to determine such routes is correct.

5 Timeline of Development

Posed Timeline of development
Sprint One (Jan 12 - 25): Focus on defining the structure of both front-end and back-end of the system. Create release planning document (Due Jan 19).
Sprint Two (Jan 26 - Feb 8): Aim to integrate APIs for location and perform calculations based on locations. Build inputting locations into the UI.
Sprint Three (Feb 9 - 22): Integrate and test the UI with the back-end of the system. Improve the UI and Test the system as needed. Create progress report One (Due Feb 23).

Sprint Four (Feb 23 - Mar 8): Add improvements to the system, such as fuel constraints, and any other kinds of route optimization.
Sprint Five (Mar 9 - 22): Focus on all kinds of tests for the system, and change the system accordingly. Create progress report 2 (Due Mar 22).
Sprint Six (Mar 23 - Apr 5): Finish the system, make any improvements, finalize UI. Prepare for final report and demonstration.